

What is claimed is:

1. A system employed by at least one web browser compatible executable application for initiating an action by a second executable application, comprising:
 - 5 a URL processor for generating, in response to a user command, a URL link in a predetermined particular format including context information and an action request identifier, the URL link representing an action to be performed by said second executable application;
 - 10 a URL link interpreter, coupled to said URL processor via a first mode of communication, for identifying said particular format URL link, for extracting said context information and action request identifier from said particular format URL link, and for initiating termination of the first mode of communication; and
 - 15 a communication processor, coupled to said URL link interpreter, for initiating communication of said extracted context information and action request identifier to said second executable application by a second mode of communication different from said first mode of communication.
2. A system according to claim 1, wherein:
 - 20 said first mode of communication comprises at least one of, (a) Internet compatible communication and (b) Hyper Text Transfer Protocol communication; and
 - 25 said second mode of communication comprises at least one of, (i) Microsoft Windows compatible message communication, (ii) socket communication, and (iii) COM communication.

3. A system according to claim 1, wherein said second executable application is a non-web enabled application.

4. A system according to claim 1, wherein:

5 said user command is received via a displayed web page; and

 said URL processor sets a location property of said browser window object to an address of said particular format.

5. A system according to claim 1, wherein:

10 said context information comprises at least one of, (a) a user identifier, (b) a patient identifier, (c) a customer identifier, (d) a source identifier, (e) a destination identifier, (f) a password, (g) a computer operational session identifier, (h) an identifier identifying said second executable application to perform said user command and (i) a data identifier; and

15 said action request identifier identifies at least one of, (i) a type of action to be performed by said second application, (ii) a type of said user command, (iii) an event and (iv) an authentication service.

20 6. A system employed by at least one web browser compatible executable application for receiving status information associated with an action performed by a second executable application, comprising:

 an interface processor for:

20

receiving application data from said second executable application in response to a user command entered via a displayed image;

5

parsing document object data associated with said displayed image to find a predetermined procedure identifier identifying an executable procedure for processing said received application data to be compatible with a web browser application; and

10

initiating execution of said identified executable procedure, in response to a command from said interface processor, to provide processed received application data to said web browser application.

7. A system according to claim 6, wherein said application data comprises at least one of, (a) a command identifier, (b) data and (c) status information associated with an action performed by said second executable application.

8. A system according to claim 7, wherein:
20 said image comprises a displayed web page processed for display by said web browser application; and

 said document object data comprises a document object model associated with said web page.

9. A system according to claim 8, wherein said web page is represented by data comprising at least one of, (a) HTML compatible data, (b) XML compatible data and (c) SGML compatible data.

5 10. A system according to claim 6, further comprising a communication processor, coupled to said interface processor, for communicating with said second executable application to acquire said application data.

10 11. A system according to claim 10, wherein said communication processor communicates with said second executable application by at least one of, (i) Microsoft Windows compatible message communication, (ii) socket communication, and (iii) COM communication.

15 12. A system according to claim 10, wherein said communication processor establishes non-polling communication with said second executable application and said second executable application provides said status information to said communication processor in response to at least one of, (i) a command by said second executable application and (ii) a request from said communication processor.

20 13. A system according to claim 7, wherein said web browser application initiates display of said received status information.

22
14. A system according to claim 7, wherein said received status information includes at least one of, (a) an identifier identifying status of performance of a task by said second executable application, (b) a status description and (c) a text message.

5

15. A system employed by a first web browser compatible executable application for initiating an action by a second executable application and for acquiring status information associated with said action, comprising:

10 a URL processor for generating, in response to a user command entered via a displayed browser image, a URL link in a predetermined particular format including context information and an action request identifier, the URL link representing an action to be performed by said second executable application;

15 a URL link interpreter, coupled to said URL processor via a first mode of communication, for identifying said particular format URL link, for extracting said context information and action request identifier from said particular format URL link, and for initiating termination of said first mode of communication;

20 a communication processor, coupled to the URL link interpreter, for initiating communication of said context information and action request identifier to said second executable application by a second mode of communication different from said first mode of communication; and

an interface processor for:

25 receiving status information associated with said action performed by said second executable application in response to said user command;

23

parsing document object data associated with said browser image to find a predetermined procedure identifier identifying an executable procedure for processing said received status information to be compatible with a web browser application; and

5

initiating execution of said identified executable procedure, in response to a command from said interface processor, to provide processed received status information to a web browser application.

10

16. A method employed by at least one web browser compatible executable application for initiating an action by a second executable application, comprising the activities of:

generating, in response to a user command, a URL link in a predetermined particular format including context information and an action request identifier, the URL link representing an action to be performed by said second executable application;

initiating communication said URL link in a first mode of communications;

20 identifying said particular format link;

extracting said context information and action request identifier from said particular format link;

initiating termination of said first mode of communication; and

initiating communication of said context information and action request identifier to said second executable application by a second mode of communication different to said first mode of communication.

17. A method employed by at least one web browser compatible executable application for receiving status information associated with an action performed by a second executable application, comprising the 5 activities of:

receiving application data associated with an action performed by said second executable application in response to a user command entered via a displayed browser image;

10 parsing document object data associated with said browser image to find a predetermined procedure identifier identifying an executable procedure for processing said received application data to be compatible with a web browser application; and

initiating execution of said identified executable procedure to provide processed received application data to said web browser application.

15

18. A method according to claim 17, further comprising the activity of updating a local storage location in response to received application data.

19. A method according to claim 18, further comprising at least one 20 of the activities of:

requesting data;

performing a procedure; and

responding to a command from a web enabled application as a result of a change in status of said web browser compatible executable application